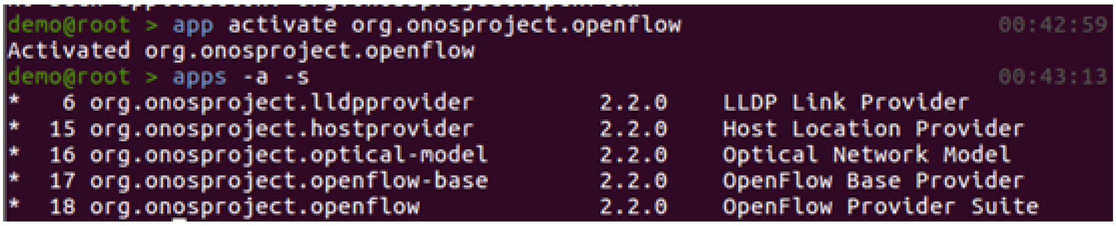
project1_109550206

軟體定義網路及網路功能虛擬化 Lab1
ID: 109550206 Name:陳品劭
Self link (https://hackmd.io/@pinchen/SDNFVproject1)

Part 1

1. When ONOS activates “org.onosproject.openflow,” what are the APPs which it also activates?

Answer:



```
demo@root > app activate org.onosproject.openflow                     00:42:59
Activated org.onosproject.openflow
demo@root > apps -a -s                                                00:43:13
*   6 org.onosproject.lldpprovider         2.2.0    LLDP Link Provider
*  15 org.onosproject.hostprovider         2.2.0    Host Location Provider
*  16 org.onosproject.optical-model        2.2.0    Optical Network Model
*  17 org.onosproject.openflow-base        2.2.0    OpenFlow Base Provider
*  18 org.onosproject.openflow             2.2.0    OpenFlow Provider Suite
```

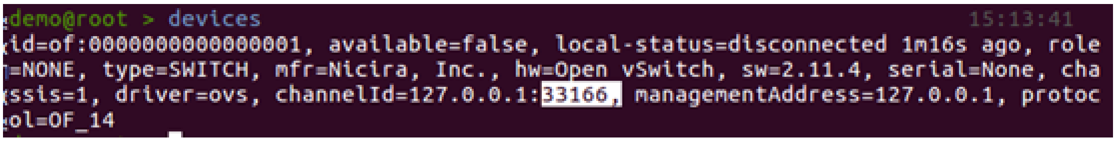
When “org.onosproject.openflow” is activated, the following APPs will also be activated.
(1) “org.onosproject.lldpprovider”
(2) “org.onosproject.hostprovider”
(3) “org.onosproject.optical-model”
(4) “org.onosproject.openflow-base”

2. As topology in p.22, can H1 ping H2 successfully? Why or why not?

Answer: No.
Since there are no flows installed on the data-plane, which forward the traffic appropriately.
We need to install forwarding flows on demand, then H1 can ping H2 successfully.
When ONOS activates a simple Reactive Forwarding app, “org.onosproject.fwd”, this problem is soved.

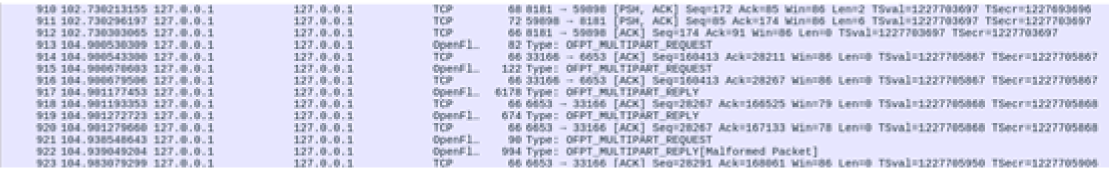
3. Which TCP port the controller listens for the OpenFlow connection request from the switch?

Answer: 6653
We can know the port of switch is 33166 by “devices”, as following picture.



```
demo@root > devices                                              15:13:41
id=of:0000000000000001, available=false, local-status=disconnected 1m16s ago, role
=NONE, type=SWITCH, mfr=Nicira, Inc., hw=Open vSwitch, sw=2.11.4, serial=None, cha
ssis=1, driver=ovs, channelId=127.0.0.1:33166, managementAddress=127.0.0.1, protoc
ol=OF_14
```

Then we let c0 ping s1 and obserce “wireshark”, we will note that the port of controler which is connection with 33166 port of switch is 6653. However we know the TCP port of ONOS is 6653.

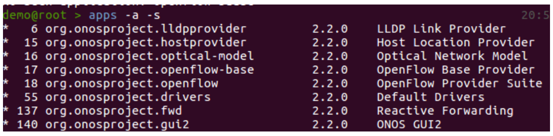


```
910 102.730213155 127.0.0.1      127.0.0.1      TCP         68 8181 → 59098 [PSH, ACK] Seq=172 Ack=85 Win=86 Len=2 TSval=1227703697 TSecr=1227693696
911 102.730296197 127.0.0.1      127.0.0.1      TCP         72 59098 → 8181 [PSH, ACK] Seq=85 Ack=174 Win=86 Len=6 TSval=1227703697 TSecr=1227703697
912 102.730303065 127.0.0.1      127.0.0.1      TCP         66 8181 → 59098 [ACK] Seq=174 Ack=91 Win=86 Len=0 TSval=1227703697 TSecr=1227703697
913 104.900530309 127.0.0.1      127.0.0.1      OpenFl…     82 Type: OFPT_MULTIPART_REQUEST
914 104.900543300 127.0.0.1      127.0.0.1      TCP         66 33166 → 6653 [ACK] Seq=160413 Ack=28211 Win=86 Len=0 TSval=1227705867 TSecr=1227705867
915 104.900670603 127.0.0.1      127.0.0.1      OpenFl…    122 Type: OFPT_MULTIPART_REQUEST
916 104.900679506 127.0.0.1      127.0.0.1      TCP         66 33166 → 6653 [ACK] Seq=160413 Ack=28267 Win=86 Len=0 TSval=1227705867 TSecr=1227705867
917 104.901177453 127.0.0.1      127.0.0.1      OpenFl…   6178 Type: OFPT_MULTIPART_REPLY
918 104.901193353 127.0.0.1      127.0.0.1      TCP         66 6653 → 33166 [ACK] Seq=28267 Ack=166525 Win=79 Len=0 TSval=1227705868 TSecr=1227705868
919 104.901272723 127.0.0.1      127.0.0.1      OpenFl…    674 Type: OFPT_MULTIPART_REPLY
920 104.901279660 127.0.0.1      127.0.0.1      TCP         66 6653 → 33166 [ACK] Seq=28267 Ack=167133 Win=78 Len=0 TSval=1227705868 TSecr=1227705868
921 104.938548643 127.0.0.1      127.0.0.1      OpenFl…     90 Type: OFPT_MULTIPART_REQUEST
922 104.939049204 127.0.0.1      127.0.0.1      OpenFl…    994 Type: OFPT_MULTIPART_REPLY[Malformed Packet]
923 104.983079299 127.0.0.1      127.0.0.1      TCP         66 6653 → 33166 [ACK] Seq=28291 Ack=168061 Win=86 Len=0 TSval=1227705950 TSecr=1227705906
```

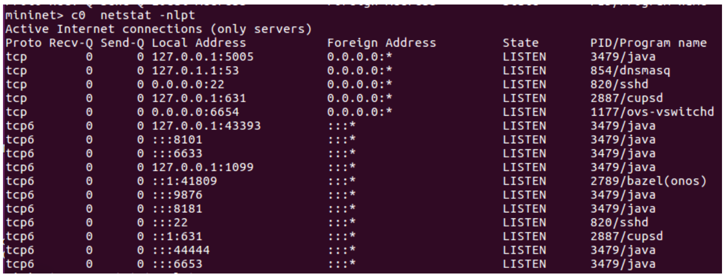
4. In question 3, which APP enables the controller to listen on the TCP port?

Answer: “org.onosproject.openflow-base”.

When no APPs deactivate, we will see these ports.



```
demo@root > apps -a -s                                            20:5
*   6 org.onosproject.lldpprovider          2.2.0    LLDP Link Provider
*  15 org.onosproject.hostprovider          2.2.0    Host Location Provider
*  16 org.onosproject.optical-model         2.2.0    Optical Network Model
*  17 org.onosproject.openflow-base         2.2.0    OpenFlow Base Provider
*  18 org.onosproject.openflow              2.2.0    OpenFlow Provider Suite
*  55 org.onosproject.drivers               2.2.0    Default Drivers
* 137 org.onosproject.fwd                   2.2.0    Reactive Forwarding
* 140 org.onosproject.gui2                  2.2.0    ONOS GUI2
```

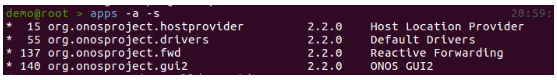


```
mininet> c0  netstat -nlpt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:5005          0.0.0.0:*               LISTEN      3479/java
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN      854/dnsmasq
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      820/sshd
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      2887/cupsd
tcp        0      0 0.0.0.0:6654            0.0.0.0:*               LISTEN      1177/ovs-vswitchd
tcp6       0      0 127.0.0.1:43393         :::*                    LISTEN      3479/java
tcp6       0      0 :::8101                 :::*                    LISTEN      3479/java
tcp6       0      0 :::6633                 :::*                    LISTEN      3479/java
tcp6       0      0 127.0.0.1:1099          :::*                    LISTEN      3479/java
tcp6       0      0 ::1:41809               :::*                    LISTEN      2789/bazel(onos)
tcp6       0      0 :::9876                 :::*                    LISTEN      3479/java
tcp6       0      0 :::8181                 :::*                    LISTEN      3479/java
tcp6       0      0 :::22                   :::*                    LISTEN      820/sshd
tcp6       0      0 ::1:631                 :::*                    LISTEN      2887/cupsd
tcp6       0      0 :::44444                :::*                    LISTEN      3479/java
tcp6       0      0 :::6653                 :::*                    LISTEN      3479/java
```

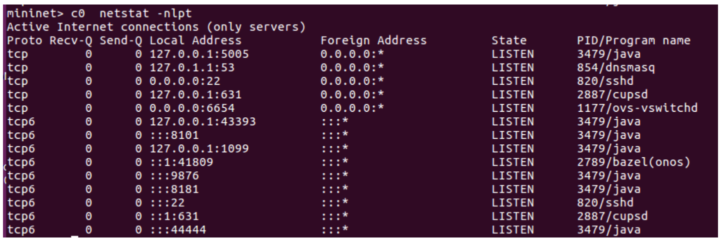
When app deactivate, there are other apps deactivate. So we need to activate them back to check which one control port open and close.

When “org.onosproject.openflow-base” deactivate, we see port 6653 and 6633 are closed. (6633好像是舊版本open flow 使用的port by網路資料)

But there are others apps are deactivated.

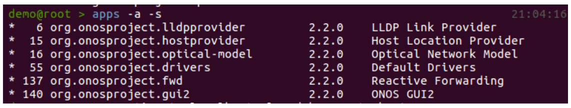


```
demo@root > apps -a -s                                            20:59:
*  15 org.onosproject.hostprovider          2.2.0    Host Location Provider
*  55 org.onosproject.drivers               2.2.0    Default Drivers
* 137 org.onosproject.fwd                   2.2.0    Reactive Forwarding
* 140 org.onosproject.gui2                  2.2.0    ONOS GUI2
```

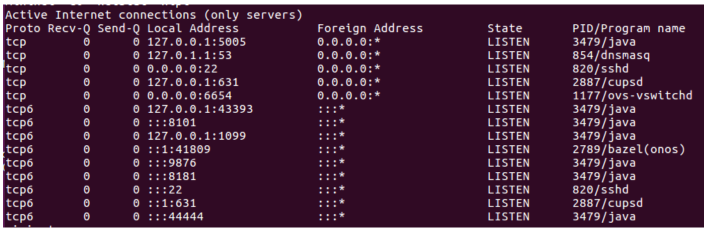


```
mininet> c0  netstat -nlpt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:5005          0.0.0.0:*               LISTEN      3479/java
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN      854/dnsmasq
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      820/sshd
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      2887/cupsd
tcp        0      0 0.0.0.0:6654            0.0.0.0:*               LISTEN      1177/ovs-vswitchd
tcp6       0      0 127.0.0.1:43393         :::*                    LISTEN      3479/java
tcp6       0      0 :::8101                 :::*                    LISTEN      3479/java
tcp6       0      0 127.0.0.1:1099          :::*                    LISTEN      3479/java
tcp6       0      0 ::1:41809               :::*                    LISTEN      2789/bazel(onos)
tcp6       0      0 :::9876                 :::*                    LISTEN      3479/java
tcp6       0      0 :::8181                 :::*                    LISTEN      3479/java
tcp6       0      0 :::22                   :::*                    LISTEN      820/sshd
tcp6       0      0 ::1:631                 :::*                    LISTEN      2887/cupsd
tcp6       0      0 :::44444                :::*                    LISTEN      3479/java
```

Then we activate others apps except openflow (since it will also activate openflow-base). Port 6653 and 6633 still close.

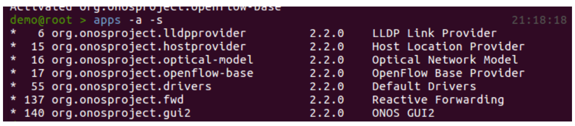


```
demo@root > apps -a -s                                            21:04:16
*   6 org.onosproject.lldpprovider          2.2.0    LLDP Link Provider
*  15 org.onosproject.hostprovider          2.2.0    Host Location Provider
*  16 org.onosproject.optical-model         2.2.0    Optical Network Model
*  55 org.onosproject.drivers               2.2.0    Default Drivers
* 137 org.onosproject.fwd                   2.2.0    Reactive Forwarding
* 140 org.onosproject.gui2                  2.2.0    ONOS GUI2
```

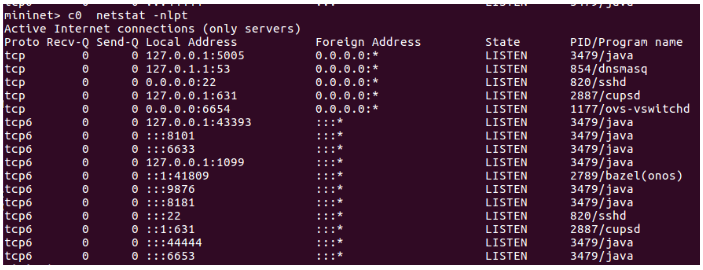


```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:5005          0.0.0.0:*               LISTEN      3479/java
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN      854/dnsmasq
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      820/sshd
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      2887/cupsd
tcp        0      0 0.0.0.0:6654            0.0.0.0:*               LISTEN      1177/ovs-vswitchd
tcp6       0      0 127.0.0.1:43393         :::*                    LISTEN      3479/java
tcp6       0      0 :::8101                 :::*                    LISTEN      3479/java
tcp6       0      0 127.0.0.1:1099          :::*                    LISTEN      3479/java
tcp6       0      0 ::1:41809               :::*                    LISTEN      2789/bazel(onos)
tcp6       0      0 :::9876                 :::*                    LISTEN      3479/java
tcp6       0      0 :::8181                 :::*                    LISTEN      3479/java
tcp6       0      0 :::22                   :::*                    LISTEN      820/sshd
tcp6       0      0 ::1:631                 :::*                    LISTEN      2887/cupsd
tcp6       0      0 :::44444                :::*                    LISTEN      3479/java
```

Then org.onosproject.openflow-base activate. (org.onosproject.openflow activate not yet). We will see port 6653 and 6633 open.



```
Activated org.onosproject.openflow-base
demo@root > apps -a -s                                                      21:18:18
*   6 org.onosproject.lldpprovider          2.2.0    LLDP Link Provider
*  15 org.onosproject.hostprovider          2.2.0    Host Location Provider
*  16 org.onosproject.optical-model         2.2.0    Optical Network Model
*  17 org.onosproject.openflow-base         2.2.0    OpenFlow Base Provider
*  55 org.onosproject.drivers               2.2.0    Default Drivers
* 137 org.onosproject.fwd                   2.2.0    Reactive Forwarding
* 140 org.onosproject.gui2                  2.2.0    ONOS GUI2
```



```
mininet> c0  netstat -nlpt
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:5005          0.0.0.0:*               LISTEN      3479/java
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN      854/dnsmasq
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      820/sshd
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      2887/cupsd
tcp        0      0 0.0.0.0:6654            0.0.0.0:*               LISTEN      1177/ovs-vswitchd
tcp6       0      0 127.0.0.1:43393         :::*                    LISTEN      3479/java
tcp6       0      0 :::8101                 :::*                    LISTEN      3479/java
tcp6       0      0 :::6633                 :::*                    LISTEN      3479/java
tcp6       0      0 127.0.0.1:1099          :::*                    LISTEN      3479/java
tcp6       0      0 ::1:41809               :::*                    LISTEN      2789/bazel(onos)
tcp6       0      0 :::9876                 :::*                    LISTEN      3479/java
tcp6       0      0 :::8181                 :::*                    LISTEN      3479/java
tcp6       0      0 :::22                   :::*                    LISTEN      820/sshd
tcp6       0      0 ::1:631                 :::*                    LISTEN      2887/cupsd
tcp6       0      0 :::44444                :::*                    LISTEN      3479/java
tcp6       0      0 :::6653                 :::*                    LISTEN      3479/java
```

However we can make sure “org.onosproject.openflow-base” control port open or close.

Part 2

Write a Python script to build the following topology:

Answer:

```python
from mininet.topo import Topo

class Project1_Topo_109550206( Topo ):
    def __init__( self ):
        Topo.__init__( self )

        # Add hosts
        h1 = self.addHost( 'h1' )
        h2 = self.addHost( 'h2' )
        h3 = self.addHost( 'h3' )

        # Add switches
        s1 = self.addSwitch( 's1' )
        s2 = self.addSwitch( 's2' )
        s3 = self.addSwitch( 's3' )
        s4 = self.addSwitch( 's4' )

        # Add links host/switch
        self.addLink( h1, s1 )
        self.addLink( h2, s2 )
        self.addLink( h3, s3 )

        # Add links switch/switch
        self.addLink( s1, s4 )
        self.addLink( s2, s4 )
        self.addLink( s3, s4 )

topos = { 'topo_part2_109550206': Project1_Topo_109550206 }
```

Creat 4 switchs:

```
sX = self.addSwitch('sX')
```

Creat 3 hosts:

```
hX = self.addHost('hX')
```

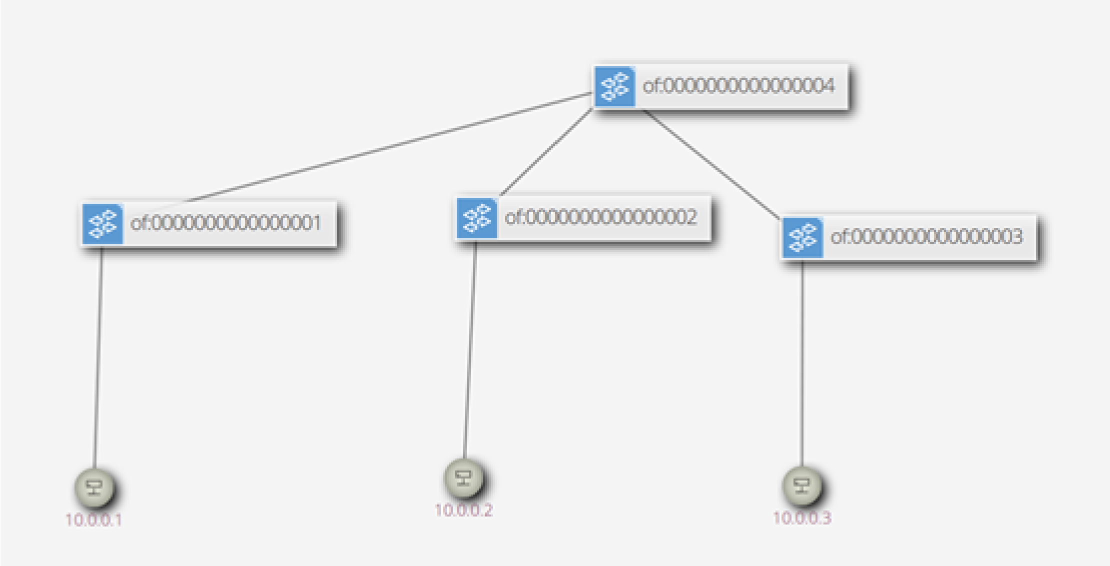
Add three link for s1, s2, s4 with s4

```
self.sddLink('sX', 's4')
```

Add three link for hX with sX

```
self.sddLink('hX', 'sX')
```

GUI:

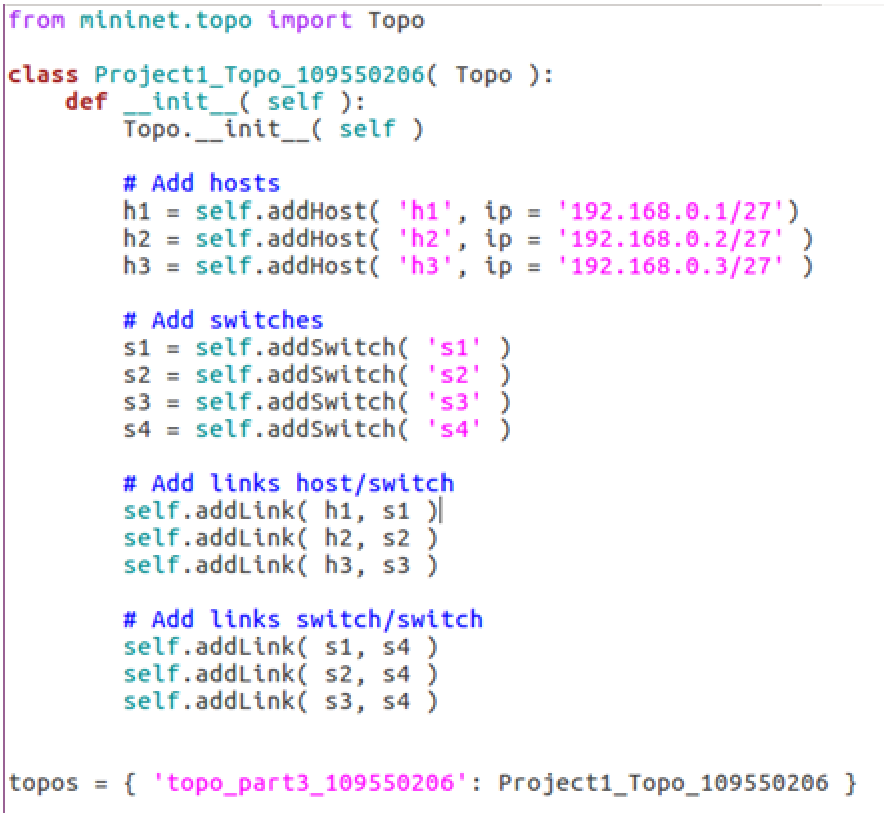


Part 3

Format for manual assignment of host IP address:

- 192.168.0.< host_number>
- netmask 255.255.255.224

Answer:



```
from mininet.topo import Topo

class Project1_Topo_109550206( Topo ):
    def __init__( self ):
        Topo.__init__( self )

        # Add hosts
        h1 = self.addHost( 'h1', ip = '192.168.0.1/27')
        h2 = self.addHost( 'h2', ip = '192.168.0.2/27' )
        h3 = self.addHost( 'h3', ip = '192.168.0.3/27' )

        # Add switches
        s1 = self.addSwitch( 's1' )
        s2 = self.addSwitch( 's2' )
        s3 = self.addSwitch( 's3' )
        s4 = self.addSwitch( 's4' )

        # Add links host/switch
        self.addLink( h1, s1 )
        self.addLink( h2, s2 )
        self.addLink( h3, s3 )

        # Add links switch/switch
        self.addLink( s1, s4 )
        self.addLink( s2, s4 )
        self.addLink( s3, s4 )


topos = { 'topo_part3_109550206': Project1_Topo_109550206 }
```

Only need to add the ip and mask when we create hosts.
ip:

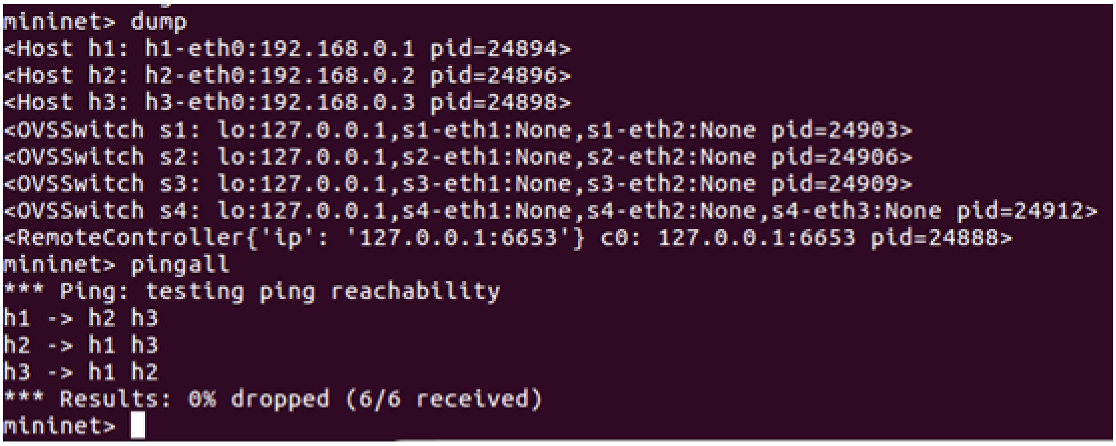
```
ip = '192.168.0.X'
```

mask need to be 255.255.255.224 = 11111111 11111111 11111111 11100000b (256 - 224 = 2^5; 32 - 5 = 27).

So we need add /27 to let others know we have 27 bits net address.

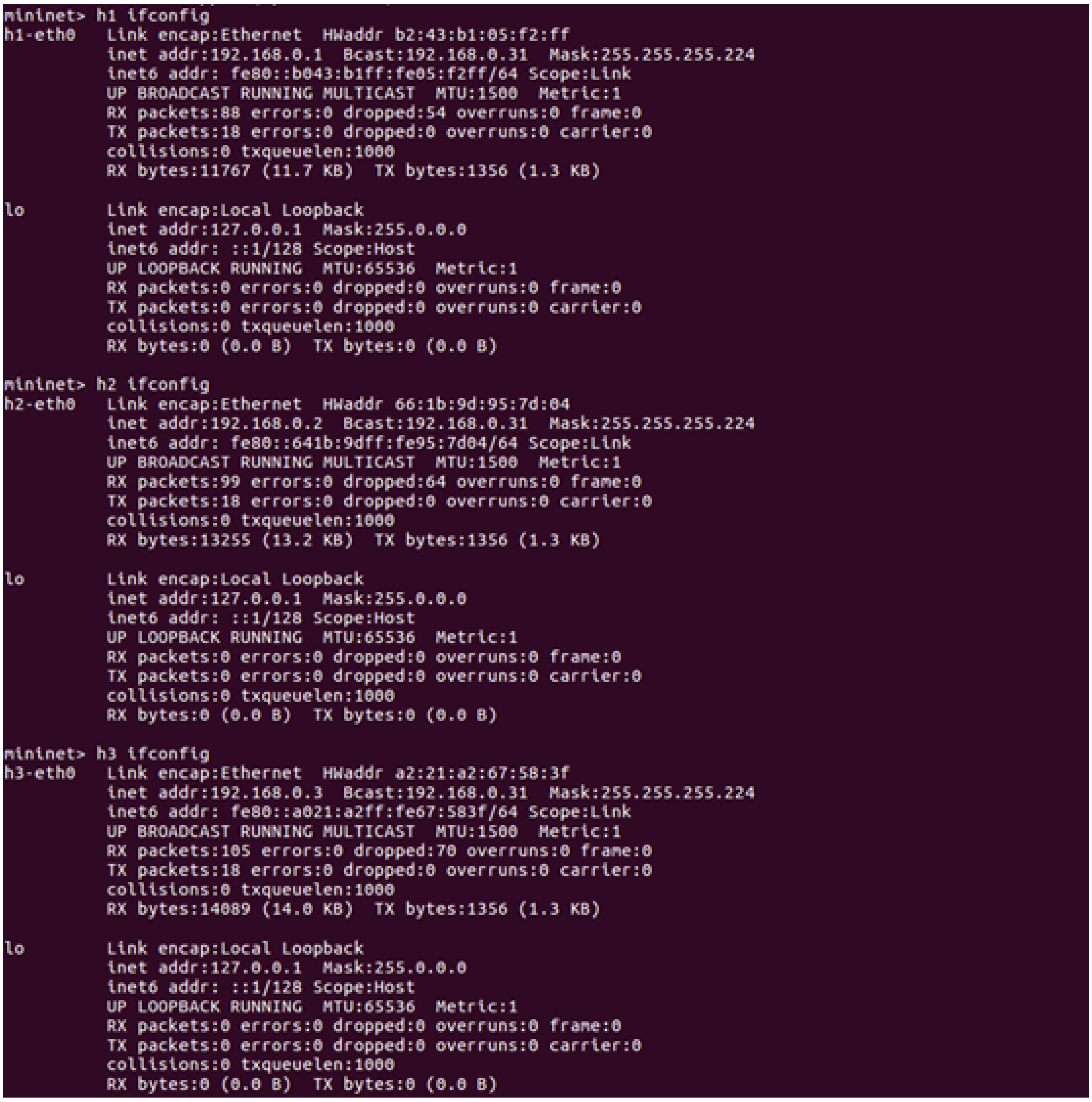
```
ip = '192.168.0.X/27'
```

Result of dump, pingall:



```
mininet> dump
<Host h1: h1-eth0:192.168.0.1 pid=24894>
<Host h2: h2-eth0:192.168.0.2 pid=24896>
<Host h3: h3-eth0:192.168.0.3 pid=24898>
<OVSSwitch s1: lo:127.0.0.1,s1-eth1:None,s1-eth2:None pid=24903>
<OVSSwitch s2: lo:127.0.0.1,s2-eth1:None,s2-eth2:None pid=24906>
<OVSSwitch s3: lo:127.0.0.1,s3-eth1:None,s3-eth2:None pid=24909>
<OVSSwitch s4: lo:127.0.0.1,s4-eth1:None,s4-eth2:None,s4-eth3:None pid=24912>
<RemoteController{'ip': '127.0.0.1:6653'} c0: 127.0.0.1:6653 pid=24888>
mininet> pingall
*** Ping: testing ping reachability
h1 -> h2 h3
h2 -> h1 h3
h3 -> h1 h2
*** Results: 0% dropped (6/6 received)
mininet>
```

Result of h1, h2, h3 ifconfig:



```
mininet> h1 ifconfig
h1-eth0   Link encap:Ethernet  HWaddr b2:43:b1:05:f2:ff
          inet addr:192.168.0.1  Bcast:192.168.0.31  Mask:255.255.255.224
          inet6 addr: fe80::b043:b1ff:fe05:f2ff/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:88 errors:0 dropped:54 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11767 (11.7 KB)  TX bytes:1356 (1.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

mininet> h2 ifconfig
h2-eth0   Link encap:Ethernet  HWaddr 66:1b:9d:95:7d:04
          inet addr:192.168.0.2  Bcast:192.168.0.31  Mask:255.255.255.224
          inet6 addr: fe80::641b:9dff:fe95:7d04/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:99 errors:0 dropped:64 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:13255 (13.2 KB)  TX bytes:1356 (1.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

mininet> h3 ifconfig
h3-eth0   Link encap:Ethernet  HWaddr a2:21:a2:67:58:3f
          inet addr:192.168.0.3  Bcast:192.168.0.31  Mask:255.255.255.224
          inet6 addr: fe80::a021:a2ff:fe67:583f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:105 errors:0 dropped:70 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:14089 (14.0 KB)  TX bytes:1356 (1.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

What you’ve learned or solved

學了mininet、ONOS的基本操作，實作網路的一些簡單架構。一開始開ONOS不知為什麼會卡在一個地方，會來重裝幾次VM後才成功，至於理由仍不明白，接著就是看基礎操作相關說明慢慢熟悉、實作，其中遇到的問題有，一時忘記可以用wireshark來得知port number，以及不知道怎麼設定子網路遮罩的問題，後來查一下才想起來。

Note of this lab (https://hackmd.io/@pinchen/SDNFVLab1)